

QUANTA

MAGAZINE

1984 TO 2010

Visit the QUANTA Website

AT

www.quanta.org.uk

Volume 27 - Issue 4
Aug/Sept 2010

Information on the Association

Membership of QUANTA, the independent QL user group, is by annual subscription. The Membership Secretary can supply full details. Copies of the association's constitution & annual accounts are available from the Secretary. There is an extensive software library. There are active national subgroups. Details are given in the Member's Guide and in this magazine.

QUANTA Committee - 2010/2011

Chairman	Sarah D. Gilpin	181, Urmston Lane, Stretford, Manchester. M32 9EH	
	Email: chairman@quanta.org.uk		Tel: 0161 865 2872
Secretary	Alison Southern	40, Distaff Road, Poynton, Cheshire. SK12 1HN	
	Email: secretary@quanta.org.uk		Tel: 01625 850067
Treasurer, Membership Secretary & Missed Issues and Acting Magazine Editor	John Gilpin	181 Urmston Lane, Stretford, Manchester. M32 9EH	
	Email: treasurer@quanta.org.uk		Tel: 0161 865 2872
	Email: membership@quanta.org.uk		
	Email: editor@quanta.org.uk		
Assistant Librarian	David Buckley	136, Denton Lane, Chadderton, Oldham. OL9 8PS	
	Email: asstlibrarian@quanta.org.uk		Tel: 0161 622 1600
Helpline, Librarian & News Editor	Dilwyn Jones	22 Erw Las, Coetmor New Road, Bethesda, Gwynedd. LL57 3NN	
	Email: helpline@quanta.org.uk		Tel: 01248 601599
	Email: librarian@quanta.org.uk		Mobile: 07749-643 922
	Email: news@quanta.org.uk		
Web Master	Dan Abbott	All contact with Web Master by email please.	
	Email: webmaster@quanta.org.uk		
Web Administrator	Keith Dunbar	44, Dalton Avenue, Stretford, Manchester. M32 9TP	
	Email: webadmin@quanta.org.uk		Mobile: 07789 483 800
Back Issues Requests	Basil Lee	31 Fairdale Gardens, London. SW15 6JW	
	Email: backissues@quanta.org.uk		Tel: 0208 789 1976

QUANTA reserves the right to publish, reject or edit any material submitted. The opinions expressed are solely those of the contributors. Member's private small ads - free (up to 50 words). Non-members private small ads (and members after 50 words) 50p per 10 words or part thereof. Terms strictly paid up-front with order. Receipt if requested. All copyrights and trademarks are hereby acknowledged.

QUANTA Magazine

CONTENTS

COPY DATE for next issue.....	3
EDITORIAL..... John Gilpin.....	4
CHAIRMAN'S NOTES..... Sarah Gilpin.....	5
COMMITTEE PRÉCIS..... Alison Southern.....	5
JOCHEN MERZ SOFTWARE _ Advert.....	6
A.G.M. NOTICE NO.1..... Alison Southern.....	7
QUANTA NEWS..... Dilwyn Jones.....	8
LIBRARY CORNER..... Dilwyn Jones.....	11
QL TODAY - Advert.....	14
QUANTA HELPLINE..... Dilwyn Jones.....	15
RWAP SERVICES - Advert.....	18
POINTS OF VIEW..... Stephen Usher.....	26
TRANSPARENCY..... George Gwilt.....	29
QUO VADIS DESIGN - Advert.....	34
RECOGNITION (MATCH)..... Steve Poole.....	35
TEN YEARS AGO..... John Gilpin.....	38
SPACE FILLING..... Bill Richardson.....	38
XCHANGE QUILL TIP..... Colin Baskett.....	39
QUANTA POSTER.....	41
HAIL! AND CONGRATULATIONS..... Colin Baskett.....	42
SUBGROUP MEETINGS.....	42

COPY DATE for OCT/NOV 2010

ISSUE is:

5TH OCTOBER 2010

Please ensure that your copy reaches the editor on or before that date. Copy on Paper or Magnetic Media or by Email.

All Contributions are Welcome.

Copy on Microdrive or Floppy Disk or CD to be accompanied by hard copy printout.

Copy by Email to - editor@quanta.org.uk

This Magazine and all the articles etc. within it are copyright © QUANTA 2009 unless otherwise stated. Reproduction and redistribution in any form is not allowed without the express written permission of the publisher. All copyrights and trademarks are hereby acknowledged.

I am absolutely astonished at the speed that time flies by between one issue of this Magazine and the next. No sooner have I posted an issue to the members, than it is time to start preparing for the next one. This is not too bad if there is some copy to work with but when there is little or no copy, then the time taken to put an issue together seems to increase exponentially.

I make no apology this month for having to look back over the last ten years or so at past issues to find something to make a publication with some interest for our members to browse through during what is left of our summer weather.

It is time, once more, for you all to decide where you would like to hold your Annual General Meeting in April 2011. For the past two years (2009 and 2010), with nobody coming forward to offer to organise and run this meeting (and hold a workshop), your committee took the middle-ground, so to speak, and chose venues in the Midlands of the UK in order that as many of you as possible were able (and prepared) to attend what I consider is the most important event of the year for QUANTA. Unfortunately, despite our efforts to be considerate to all members alike, the attendance at the A.G.Ms. in the last two years has been very low.

It has been said on more than one occasion that QUANTA is more likely to die from apathy than from any other single cause. Is that what you all want? I hope not! But only time will tell.

I understand that there are still 16 members who have not yet renewed their annual subscription, for the JULY membership period, despite being sent the usual reminders. As this figure represents about 8.5% of our remaining membership, I hope that they reconsider their decision not to continue with QUANTA.

Once again, I will have to stand down from the QUANTA Committee at next year's A.G.M. and this year, the committee will not have the option of co-opting me back again - I must stand down for a minimum of one year - is anyone out there prepared to share the load with the rest of my colleagues?

The answer is in your hands. Let's see who has the initiative to take it on.

The year rolls on, the summer passes and the time has come to think of the Annual General Meeting 2011. Our venues in 2009 and 2010 have been in the Midlands to try and make the event accessible to more of our members. The first A.G.M. notice is included in this magazine to see whether any subgroup is prepared to host the A.G.M. in a place that is easily accessible by all means of transport, and has access to those carrying equipment etc. There is the option of holding the A.G.M. in Birmingham in the same hall as 2010. ANY VOLUNTEERS??

Progress with the member's section of the QUANTA website has been slow, partially due to heavy work commitments of our two Web experts. I am hoping that a push will be made with this over the next month or so.

Activity on all fronts has been slow over the last few weeks, with little material received by our editor for this magazine.

We have recently received a large donation of equipment and books from a NEMQLUG member, John Butterworth, and QUANTA have donated a nominal sum to a charity of John's choice - The British Heart Foundation. John has been a long term member of NEMQLUG, but has recently been unable to attend through ill-health. John is now almost entirely house-bound and has decided to transfer all his computer energy to a PC and hopefully a QL emulator. QUANTA wishes all the best to John and his wife, Alice. John Gilpin is delivering much of the equipment and books to Rich Mellor for him to put on his website. Our thanks go to both John Butterworth and Rich Mellor for this potential boost to our funds.

John Gilpin is making slow but steady progress in scanning the early magazines into electronic format. He is still looking for volunteer proof-readers, please.

COMMITTEE PRÉCIS

Alison Southern

Family circumstances have meant that John and I have had very little contact with the QL world over the past few months. I know the rest of the committee have been working tirelessly in the meantime and we all have much to thank them for. The time is rapidly approaching when we will be asking for your nominations for next year's committee, please will you all consider the possibility of helping out. Without the involvement of the members QUANTA will die, like so many clubs and societies have done over the past few years. We have passed the age of 25 and are still going, but whether we make 35 is in your hands. On a brighter note, have you visited the QUANTA website recently? It is well worth a visit and, of course, all feedback will be gratefully received.

Jochen Merz SOFTWARE

Kaiser-Wilhelm-Str. 302
47169 Duisburg, Germany

Fax +49 203 502012
Email: SMSQ@J-M-S.com

QPC2 Version 3 + SMSQ/E Software QL-Emulator for PC's		EUR 59,90
QPC2 Version 3 - Upgrade from QPC2 Version 2		EUR 19,90
QPC2 Version 3 - Upgrade from QPC2 Version 1		EUR 39,90
SMSQ/E ATARI or (Super)GoldCard or QXL		EUR 39,90
QPC Print - printer emulation driver for QPC		EUR 39,90
Agenda Agenda program for WMAN and Prowess	[V1.09]	EUR 14,90
Success Database front-end for WMAN	[V2.05]	EUR 19,90
QD2003 Pointer-Environment-Editor	[VB.01]	EUR 39,90
QD2003 Upgrade from QD98	[VB.01]	EUR 9,90
QD2003 Upgrade from previous versions	[VB.01]	EUR 19,90
QMAKE Pointer-driven MAKE for GST/Quanta Assembler	[V4.31]	EUR 14,90
BASIC Linker	[V1.21]	EUR 14,90
WINED Floppy/Harddisk Sector- & File-Editor	[V1.26]	EUR 14,90
FIFI II File-Finder - Extremely useful	[V4.31]	EUR 14,90
FIFI II Upgrade from Fifi V1, 2 or 3	[V4.31]	EUR 9,90
EPROM Manager	[V3.02]	EUR 14,90
QSpread2003 Spreadsheet Program	[V4.04]	EUR 39,90
QSpread2003 Upgrade from QSpread2001	[V4.04]	EUR 9,90
QSpread2003 Upgrade from V1	[V4.04]	EUR 29,90
QPAC I Utility programs	[V1.11]	EUR 19,90
QPAC II Files, Jobs & other Things	[V1.45]	EUR 29,90
QTYP II Spell checker	[V2.17]	EUR 24,90
QPTR Pointer Toolkit	[V0.30]	EUR 39,90
DISA Interactive Disassembler	[V3.04]	EUR 29,90
typeset-ESC/P2 text87 driver for all ESC/P2 printers (incl. Stylus)		EUR 24,90
CueShell	[V2.14]	EUR 29,90
CueShell for QPC	[V2.14]	EUR 20,00
SER Mouse software mouse driver for serial mice		EUR 10,00
EasyPTR Version 4	[V4]	EUR 59,90
EasyPTR Version 4 - Upgrade from earlier versions	[V4]	EUR 39,90
text87plus4patch - now for QPC, QXL, Q40, Q60, Aurora		EUR 10,90
QDT - QL Desktop program		EUR 59,90
QMENU Version 8 - New! with new, printed Manual	[VB.02]	EUR 24,90
QMENU Version 8 - Update from earlier Versions, also with printed manual		EUR 17,90
QMENU Version 8 - New/Update for QL Today subscribers, with prtd manual		ONLY EUR 14,90

Please add EUR 4,50 for postage EUROPE, or EUR 6,50 for postage REST OF WORLD

We accept VISA, MasterCard & Diners Club online and offline! Amex only by mail or fax, not email!

New payment methods for our customers: Money transfer to "local" account in many countries!

- Deutschland: Jochen Merz, Account 493 50 431, Postbank Essen, BLZ 360 100 43
- Österreich: Jochen Merz, Account 85055317, PSK Wien, BLZ 60000
- Switzerland: Jochen Merz, Account 60 690080-4, PostFinance, Clearing Nr: 09000
- The Netherlands: Jochen Merz, Gronummer 3258439, Postbank NL, Amsterdam
- and from all other countries in EUR with IBAN and BIC to account
Jochen Merz, Deutsche Postbank AG, IBAN DE21 3601 0043 0611 1004 37 / BIC: PBNKDEFF 360
- UK customers can pay in £ (convert EUR prices above to £ by multiplying with 0.90) to
Jochen Merz, Account 83795395, Citibank UK, Sort code 30-00-45
or send cheques in £ - no fee for UK sterling cheques!
- US customers can pay in US\$ (convert EUR prices above to US\$
by multiplying with 1.35) - no fee for US cheques in US\$!

Cheques payable to Jochen Merz only!

Price list valid until 30th of Sept. 2010

QUANTA Magazine

Annual General Meeting Notice No. 1

Invitations are invited from QUANTA Members or Sub-Groups to organise and run a workshop at which the Annual General Meeting can be held during the second half of April 2011.

All the usual criteria should be considered but not restricted to the following:

- Adequate off road car parking
- Easy access to the building (and/or meeting room(s))
- Availability of space and equipment to provide refreshments
- Room large enough to meet the requirements of members attending
- Adequate Power outlets and tables & chairs for all members to bring their own equipment
- Separate adjacent room that can be used for workshop talks and presentations as well as the Annual General Meeting (Normally held between 2.00 pm and 4.00 pm).

Anyone or any Sub-group who wishes to be considered for organising this event should write to the Secretary with full details and costings to be received not later than noon on **1st October 2010**. The person or Sub-Group chosen for this event will be offered the full support of the committee in preparing this event.

Signed: Alison Southern, (on behalf of the QUANTA Committee).

QUANTA Secretary

Secretary@quanta.org.uk - or Postal address inside the front cover of this Magazine.

QL Videos On YouTube

Anton Preinsack has re-released his video of the QL meeting in Austria, with improvements to the sound in one scene and reduced level of music.

He has also made an update of the video with some corrections and a new scene (at 3:35) where you can see the menus of QEmulator for MacOSX.

The new video can be seen at:

<http://www.youtube.com/watch?v=O4Q7IUMKvRM>

Urs König also reports:

"Great to see more and more QL videos on YouTube.

YouTuber d3x128 from the Czech Republic uploaded recently a 4 minutes video covering Eidersoft mICE on the Sinclair QL.

Watch it at:

<http://www.youtube.com/watch?v=B5CAyF7yjVs>

I'm currently uploading some more QL videos to YouTube. So, stay tuned."

Well, I did. And here's some further news from Urs:

"It has been a while since aftermath II (April) of the "QL & Mac are 25" show which was held on Oct 31-Nov 1, 2009. The remaining hard job to do was (in fact it still is) the video editing and cutting.

Well, after some other folks uploaded some nice Sinclair QL related videos recently I got motivated to do my job and started editing

QUANTA Magazine

and cutting of the raw footage I've collected at the show last autumn.

My recent "QL & Mac are 25" show uploads are:

"The Animated Head" 3D demo 1988

<http://www.youtube.com/user/QLvsJaguar#p/u/13/40ByKsPB9vc>

QTop desktop suite 1989 (pre Windows 3.0) - part 1 of 2

<http://www.youtube.com/user/QLvsJaguar#p/u/15/rbFMDw1MOXw>

QTop desktop suite 1989 (pre Windows 3.0) - part 2 of 2

http://www.youtube.com/user/QLvsJaguar#p/u/16/zVZICdYj_38

QL ELITE 3D demo 1991

<http://www.youtube.com/user/QLvsJaguar#p/u/17/eYoTSGvWf78>

URL of the show web page is:

<http://tinyurl.com/ql-mac-show>

I also uploaded two videos taken at the 2010 QL meeting in Austria which was at beginning of June.

SMS2 OS (Sinclair QL QDOS successor) on ATARI STE

<http://www.youtube.com/user/QLvsJaguar#p/u/19/kY6LdsW-fTE>

Sinclair QL KARATE game on a Q40

<http://www.youtube.com/user/QLvsJaguar#p/u/18/d11fQvqZECM>

Don't forget to have another look at Linus Torvalds (the creator of the Linux kernel) GMOVE software blitter for the QL on the QL-is-25 web page. URL is:

<http://tinyurl.com/ql-is-25>

Enjoy the videos and the summer!"

QemuLator News

Daniele Terdina reports on development of the OSX (Mac) version of QemuLator:

"Development is continuing. The OS X version that was seen at the Vienna meeting was close to complete (I still need to write the manual and do some testing, though). On Windows, all new features are done, but I'm currently rewriting the Windows 7 version of full screen emulation."

QMAC Assembler

Following on from the news in the last issue that Urs König has secured the free release of old GST QL software, QUANTA is also pleased to announce that the QUANTA version of the macro assembler, QMAC, has also been released as freeware.

QMAC version 1.06 and Linker v1.03 are available - members can get them free from the library on a 2 disk set - UG30 - including manuals as PDF files. This is the QUANTA version modified by Phil Borman. Sadly, Phil does not have the sources any more and we tried to contact various people we could think of who may have held copies at the time, without any luck. Phil did say that if copies of his modified sources could be found, he would have no objection to them being made PD.

So, here is a public appeal... if anyone holds copies of the sources to this assembler, please get in touch with a member of QUANTA committee, as it would be helpful if the sources were available to update should the need ever arise.

Library disk UG 29

UG29 has been updated with several new programs and several sample QL font files. In addition to the font viewer program already on this disk, the following have been added to this disk during July:

Character Pick - displays the whole QL font and lets you left click ("Hit") on a character to display its CODE, or right click ("Do") on a character to send it to the Stuffer Buffer, from where it can be sent to another program with the key press ALT-SPACE. A little time saving utility which means you don't have to look up the key press combinations needed for accented characters and symbols not marked on the keyboard. Needs pointer environment.

Font Editor - a pointer driven font editor for QL fonts. Works on Window Manager 2 systems (which means SMSQ/E version 3.00 or later, or QDOS pointer environment version 2.00 or later). Includes commands to *italicise* or **bolden** characters, copy characters, save and load fonts, merge fonts (to change character ranges). Can handle standard QL 5 or 6 pixel wide fonts for all character sizes, and 8 pixel wide fonts for use in CSIZE width 1 only.

Euro Font For QDOS - for some time now SMSQ/E has included a Euro currency symbol at character code 181 (CTRL SHIFT U). This font file is for use on QDOS systems, to provide a Euro symbol at the same position in the character set.

Proportional Spacing - Two BASIC programs, an article and two pairs of fonts showing how to implement a rudimentary proportional printing system for the screen, using standard QL fonts. This allows characters to have different widths - for example the letters W, M and O can be larger than other letters, rather than the simple monospaced QL font system.

QUANTA Magazine

Sample Fonts - a collection of several QL screen fonts, including some 8 pixel wide fonts designed for use in CSIZE 1,0 or CSIZE 1,1.

Library disk UG30

UG30 is a new 2 disk set comprising the QUANTA QMAC assembler. Following on from the news in the last issue that GST have kindly allowed their QL software to be made available as freeware (they retain copyright, but allow the software to be supplied free of charge or on a non-profit basis), QUANTA have also allowed the QUANTA version of the QL Macro Assembler (enhanced by Phil Borman) to become available free of charge in line with GST's decision. This includes QMAC version 1.06 and linker version 1.03. Disk 2 contains the QMAC, Linker and Screen Editor manuals as .PDF files.

Library Disks SP56 and SP57

Don't forget than Tony Hill's searchable PDF version of the QL manual has recently become available from the library. Disk SP56 contains the trial version, a cut down version needing a password of "join-quanta" to use, but which can be circulated to non-members, while SP57 contains the full version needing no password, but which is available to QUANTA members only.

Magazine Listings ML01

This disk continues to be updated with listings from the QUANTA magazine, covering the period from October 2005 to the present day. Some listings are available on the QUANTA website, but disk ML01 will continue to be available for those unable to download the QL program listings from the website.

QUANTA Magazine

HANOI Program, disk GG01

A minor change has been made to this program, to allow it to run on SMSQ/E systems. If you already have a copy, please add the following line to the program, to add an End Define statement to the Get_Move procedure without a termination:

```
1175 END DEFine
```

Library On Website

Work continues on making the QUANTA Library available to members via the QUANTA website. This has taken much longer than anticipated due mainly to the need to fully and properly implement the Content Management System (CMS) for the website, and to set up and test a password protected database system for selecting and downloading the library disk required using the CMS-based website. I very much hope that this will not take too much longer before becoming available.

Ordering Programs From The Library

For a full guide to ordering programs from the QUANTA Library, please refer to Section C of your QUANTA Member's Guide.

A list of the programs in the library is contained in the Library Guide, which is already available to download from the QUANTA website at the URL

<http://www.quanta.org.uk/index.php?id=44>

QL Today Volume 15

Customer-Nr.

Name: _____
 Street: _____
 Town: _____
 City: _____
 Country: _____
 EMail: _____

Subscriptions are also available online from Quo Vadis Design in the UK www.ql-qvd.com
 Full subscription details are also available from www.qltoday.com

I hereby subscribe to **QL Today** for 4 issues of Volume 15. The total price for all four issues is as follows, including postage and packing (depending on destination)

Destination	price for Volume 15
Germany, Netherlands & United Kingdom	EUR 29.90
Rest of Europe	EUR 31.90
Rest of World (airmail)	EUR 42.90

Same price as last year!

Please charge my credit card: VISA MasterCard Diners Club
 Expires -
 Card Verification Code:

Money transfer to one of the following accounts:
 Deutschland: Jochen Merz, Account 493 50 431, Postbank Essen, BLZ 360 100 43
 Österreich: Jochen Merz, Account 85055317, PSK Wien, BLZ 60000
 Switzerland: Jochen Merz, Account 60-690080-4, PostFinance, Clearing-Nr. 09000
 The Netherlands: Jochen Merz, Gironummer 3258439, Postbank/ING NL Amsterdam
 and from all other countries in EUR with IBAN and BIC to account
 Jochen Merz, Deutsche Postbank AG, IBAN: DE21 3601 0043 0611 1004 37 / BIC: PBNKDEFF 360
 UK customers can pay £26.90 (valid until July 2010) to
 Jochen Merz, Account 83795395, Citibank UK, Sort code 30-00-45
 or send cheques in £ - no fee for UK sterling cheques (payable to *Jochen Merz* only!)

Paypal: Log into your paypal account and send the money in EUR to paypal@J-M-S.com

Date, Signature _____

Please fill in and send to Jochen Merz Software, Kaiser-Wilh.-Str. 302, 47169 Duisburg, Germany, or Fax to +49 203 501517 or scan & Email to SMSQ@J-M-S.com

Issue 4 of QL Today Volume 14 was ready earlier than scheduled to be available at the Vienna Show - again with 25% extra pages! Back issues with extremely interesting Tony Tebby articles available! Become part of Volume 15 - subscribe now!

Members wishing to submit helpline requests via email can use the email address helpline@quanta.org.uk or if you prefer to use traditional post, please send the helpline request to me via the address printed inside the front cover of the newsletter.

Obviously, we cannot guarantee to answer every query we receive, but we will do our best! Where we have been unable to answer the queries, we may print the help request as an open request in the newsletter to ask if any of the readers can come up with a solution. And, of course, if readers feel that they have a better solution than we came up with, or would like to correct any errors we make, please write to us!

_Sav Files

A member on the ql-users mailing list recently raised the issue of decoding tokenised SBASIC programs. These are BASIC program files, saved with the **QSAVE** command, which are designed to load faster than ordinary plain text BASIC programs saved with the ordinary SAVE command. When I looked into it, I found there wasn't really much by way of publicly available documentation on the **_sav file** format.

QSAVE works by saving the program in pretty much the same way as it is stored in memory, with the procedures, functions, extensions and keywords all stored as tokens. This lets the BASIC interpreter load the program without having to "tokenise" the plain text version of the program - it saves a lot of processing and is particularly effective on slower systems.

QSAVE and **QLOAD** extensions were originally supplied as a small software package by Liberation Software (publishers of the Qliberator compiler) for the original QL. LOADING SuperBASIC programs was quite slow with long programs on a QL, so Liberation Software released this handy little utility package, which also included a utility called **QREF** for listing names of variables, procedures, functions, etc. Lots of us still use it today!

A **QSAVE _sav** file has a table header which includes some name table information at the beginning of the file, which includes information such as the number of name table entries for this program, the name table length and the number of lines in the BASIC program. Then comes the name table itself - the list of names used by the program. The rest of the file is basically a

copy of the BASIC program in a very similar format to that in which it is stored in memory.

Acknowledgements

I am very grateful to Per Witte and Norman Dunbar who sent me all sorts of useful information to help me understand the file format. Both sent me programs to help with decoding the programs. Norman wrote a program many years ago to decode these **_sav** files and as he hadn't updated it for SBASIC, I tentatively volunteered to update it. As it happened, it had been written so well originally that all I had to do was make fairly small changes to allow it to handle two new facilities in SBASIC (integer and hex constants) and change a couple of variable names to avoid clashes with new extension names in SBASIC. Norman kindly gave me permission to publish the new listing. There's also a wealth of information in the Jan Jones book, "QL SuperBASIC - The Definitive Handbook" in Appendix C which lists the tokens used to store a SuperBASIC program.

Tokens

The tokens are represented by special values in memory. For example, the keyword "REMark" is not stored as the word REMark, but rather as a two byte value, hex 811e.

Now this sounds awfully complex, and in one way it is - you have to be able to understand hex numbers to follow much of this, as you have to follow a trail of hexadecimal values to decode a tokenised SuperBASIC program. Or SBASIC program - the basic file format is essentially the same, just that SBASIC has added a few new tokens for binary and hexadecimal constants using the % and \$ prefixes for values, e.g. %1111 or \$F to represent the decimal value 15.

Don't worry too much if you don't understand all this - you can still use the program and try to follow how it is decoding the file. This might be useful to those who might like to use it to help them write a program to handle and manipulate these programs, e.g. advanced users might like to try to write a program which copies and pastes whole routines, or builds libraries of routines, a kind of basic development environment, even!

QUANTA Magazine

The tokens which represent the various keywords, operators, separators, names and so on all start with \$8xxx. The second "nybble" in the hex value indicates the group or type of name:

\$80xx indicates number of running spaces

\$81xx keywords

\$82xx unused

\$83xx unused

\$84xx symbol identifier

\$85xx operator

\$86xx monadic operators

\$87xx unused

\$8800 xxxx Names - Number of entry in name table

\$89xx unused

\$8Axx unused

\$8B..... strings

\$8C00... text

\$8D00xxxx Line number

\$8Exx Separators

\$8F... Floating point value, top 4 bits indicate if a float, binary, or hex representation

These are explained in more detail in the Jan Jones book, but also quite easy to follow in Norman's listing below.

The procedure called `Decode_Header` decodes the short header with name table information. It tries to identify the file as a `_sav` file by looking at the first four bytes of the file in line 310, but two of these bytes can vary depending on how old the particular file version is - if you know of any other values to check for here, let me know! Then, the `Decode_Name_Table` routine wades through the name table list, and finally the `Decode_Program` routine steps through the program changing it back to plain text. Which is essentially what this program does - decodes a `_sav` file into an ordinary untokenised BASIC program.

The program is well commented (thanks, Norman!) to help you follow what it does.

After the `Decode_Program` routine comes a series of routines such as `Multi_Spaces` and `Keywords` which show how to handle the various tokens. At the end of the listing comes an initialisation routine which has a list of the keywords, operators and separators corresponding to token values.

Sinclair QL Re-Releases RWAP Services

We have been working hard to safeguard the future of the Sinclair QL through beginning to compile a repository of QL titles.

The repository enables people to see what software was produced for the Sinclair QL and also to be able to purchase working copies when their microdrive cartridges fail. Where we can, we obtain permission from the copyright holder to re-release items on the market.

This has enabled us to bring some of the titles back to the market as well as re-kindle interest in the Sinclair QL and its emulators.

Old favourites brought back to life include:

- Jungle Eddi
- The Lost Pharaoh
- Pengi
- Microdeal's Flight Simulator
- Microdeal's QL Hopper
- Microdeal's The King
- Cuthbert in Space

We still retain our existing catalogue, including:

- Open Golf
- Stone Raider II
- Hoverzone
- Deathstrike
- Flightdeck

All games are available on disk or for use with Q-emulator on the PC for **ONLY £5 each.**

Microdrive versions also available - £10 each

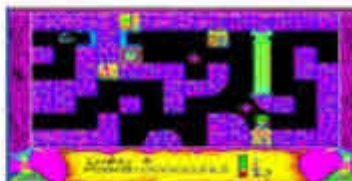
Visit the QL Wiki for more details on software, books and hardware for the Sinclair QL:
www.rwapadventure.com/ql_wiki/

RWAP Software
3 Dale View Court
Fulford
Stoke-on-Trent
Staffordshire
ST11 9BA
Tel: 01782 398143
Email: rich@rwapsoftware.co.uk
Website:
www.rwapsoftware.co.uk
or sellmyretro.com

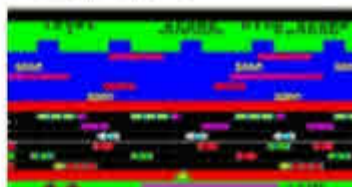
Working to safeguard the QL for the future www.rwapsoftware.co.uk



QL Pengi by Jochen Mezz



The Lost Pharaoh by Talent



QL Hopper by Microdeal



The King by Microdeal



Cuthbert in Space by Microdeal

QUANTA Magazine

```
100 REMark _SAV file decoder
110 :
120 CLS
130 PRINT 'SAV File Decoder'\
140 INPUT 'Which _sav file ? ';sav$
150 IF sav$ = '' THEN STOP: END IF
160 :
170 initialise
180 decode_header
190 IF NOT _quit
200     decode_name_table
210     decode_program
220 END IF
230 RELEASE_HEAP float_buffer
240 CLOSE #3
250 :
260 DEFine PROCedure decode_header
270     LOCAL head$(4), name_table_length
280     _quit = 0
290     OPEN_IN #3,sav$
300     head$ = FETCH_BYTES(#3, 4)
310     IF (head$ <> 'Q1' & CHR$(0) & CHR$(0)) AND (head$
<> 'Q1' & CHR$(2) & CHR$(192)) AND (head$ <> 'Q1' &
CHR$(3) & CHR$(128))
320     PRINT head$, head$(1); head$(2)!! CODE(head$(3))!
CODE(head$(4))\
330     PRINT sav$ & ' is not a SAV file, or has a new
flag.'
340     CLOSE #3
350     _quit = 1
360     RETURN
370 END IF
380 name_table_entries = GET_WORD(#3)
390 name_table_length = GET_WORD(#3)
400 program_lines = GET_WORD(#3)
410 max_name_size = name_table_length - (4 *
name_table_entries) / name_table_entries
420 :
430 PRINT sav$
440 PRINT 'Number of name table entries : ';
name_table_entries
```

QUANTA Magazine

```
450 PRINT 'Name table length           : ' ;
name_table_length
460 PRINT 'Number of program lines     : ' ;
program_lines
470 PRINT
480 :
490 DIM name_table$(name_table_entries -1,
max_name_size)
500 float_buffer = RESERVE_HEAP(6)
510 _quit = (float_buffer < 1)
520 END DEFine decode_header
530 :
540 DEFine PROCedure decode_name_table
550 LOCAL x, name_type, line_no, name_length, name$,
lose_it$(1)
560 LOCAL num_procs, num_fns
570 num_procs = 0
580 num_fns = 0
590 FOR x = 0 TO name_table_entries -1
600 name_type = GET_WORD(#3)
610 line_no = GET_WORD(#3)
620 name_length = GET_WORD(#3)
630 name$ = FETCH_BYTES(#3, name_length)
640 IF name_length && 1
650 lose_it$ = INKEY$(#3)
660 END IF
670 IF name_type = 5122 THEN num_procs = num_procs
+ 1
680 IF name_type >= 5377 AND name_type <= 5379
690 num_fns = num_fns + 1
700 END IF
710 PRINT x; ' Name type = ' ; HEX$(name_type, 16) &
' ' ;
720 PRINT 'Line number = ' ; line_no & ' ' ;
730 PRINT 'Name length = ' ; name_length; ' ' ;
740 PRINT 'Name = <' & name$ & '>'
750 name_table$(x) = name$
760 END FOR x
770 PRINT 'There are ' & num_procs & ' PROCs'
780 PRINT 'There are ' & num_fns & ' FNs'
790 END DEFine decode_name_table
```

QUANTA Magazine

```
800 :
810 :
820 DEFine PROCedure decode_program
830   LOCAL x, type_byte, program_line
840   :
850   REMark WORD = size change
860   REMark LONG = $8D00.line number
870   REMark rest of line
880   :
890   REPEAT program_line
900     IF EOF(#3) THEN EXIT program_line: END IF
910     line_size = line_size + GET_WORD(#3)
920     IF line_size > 65536 THEN line_size = line_size
- 65536: END IF
930     IF GET_WORD(#3) <> HEX('8d00')
940       PRINT 'Program out of step.'
950       CLOSE #3
960       STOP
970     END IF
980     PRINT GET_WORD(#3); ' ';
990     line_done = 0
1000    REPEAT line_contents
1010      type_byte = CODE(INKEY$(#3))
1020      SElect ON type_byte
1030        = HEX('80'): multi_spaces
1040        = HEX('81'): keywords
1050        = HEX('84'): symbols
1060        = HEX('85'): operators
1070        = HEX('86'): monadics
1080        = HEX('88'): names
1090        = HEX('8B'): strings
1100        = HEX('8C'): text
1110        = HEX('8E'): separators
1120        = HEX('D0') TO HEX('DF') : floating_points
1 : REMark % binary number
1130        = HEX('E0') TO HEX('EF') : floating_points
2 : REMark $ hex number
1140        = HEX('F0') TO HEX('FF') : floating_points
3 : REMark floating point
1150      END SElect
1160    IF line_done THEN EXIT line_contents: END IF
```

QUANTA Magazine

```
1170     END REPeat line_contents
1180     END REPeat program_line
1190 END DEFiNe decode_program
1200 :
1210 :
1220 DEFiNe PROCedure multi_spaces
1230   :
1240   REMark $80.nn = print nn spaces
1250   :
1260   PRINT FILL$( ' ', GET_BYTE(#3));
1270 END DEFiNe multi_spaces
1280 :
1290 :
1300 DEFiNe PROCedure keywords
1310   :
1320   REMark $81.nn = keyword$(nn)
1330   :
1340   PRINT keyword$(GET_BYTE(#3)); ' ';
1350 END DEFiNe keywords
1360 :
1370 :
1380 DEFiNe PROCedure symbols
1390   LOCal sym
1400   :
1410   REMark $84.nn = symbol$(nn)
1420   :
1430   sym = GET_BYTE(#3)
1440   PRINT symbol$(sym);
1450   line_done = (sym = 10)
1460 END DEFiNe symbols
1470 :
1480 :
1490 DEFiNe PROCedure operators
1500   :
1510   REMark $85.nn = operator$(nn)
1520   :
1530   PRINT operator$(GET_BYTE(#3));
1540 END DEFiNe operators
1550 :
1560 :
1570 DEFiNe PROCedure monadics
```

QUANTA Magazine

```
1580  :
1590  REMark $86.nn = monadic$(nn)
1600  :
1610  PRINT monadic$(GET_BYTE(#3));
1620  END DEFine monadic
1630  :
1640  :
1650  DEFine PROCedure names
1660  LOCal ignore
1670  :
1680  REMark $8800.nnnn = name_table$(nnnn)
1690  :
1700  ignore = GET_BYTE(#3)
1710  ignore = GET_WORD(#3)
1720  IF ignore > 32768 THEN ignore = ignore - 32768:
END IF
1730  PRINT name_table$(ignore);
1740  END DEFine names
1750  :
1760  :
1770  DEFine PROCedure strings
1780  LOCal delim$(1), size
1790  :
1800  REMark $8B.delim.string_size = 'delim'; string;
'delim'
1810  :
1820  delim$ = INKEY$(#3)
1830  size = GET_WORD(#3)
1840  PRINT delim$; FETCH_BYTES(#3, size); delim$;
1850  IF size && 1
1860      size = GET_BYTE(#3)
1870  END IF
1880  END DEFine strings
1890  :
1900  :
1910  DEFine PROCedure text
1920  LOCal size
1930  :
1940  REMark $8C00.size = text
1950  :
1960  size = GET_BYTE(#3)
```

QUANTA Magazine

```
1970 size = GET_WORD(#3)
1980 PRINT FETCH_BYTES(#3, size);
1990 IF size && 1
2000     size = GET_BYTE(#3)
2010 END IF
2020 END DEFine text
2030 :
2040 :
2050 DEFine PROCedure separators
2060 :
2070 REMark $8E.nn = separator$(nn)
2080 :
2090 PRINT separator$(GET_BYTE(#3));
2100 END DEFine separators
2110 :
2120 :
2130 DEFine PROCedure floating_points (fp_type)
2140 REMark modified for % and $ SBASIC values
22.01.10 - DJ
2150 LOCAL number$(6), fpt
2160 fpt = fp_type : REMark to avoid SEL ON last
parameter issue later
2170 :
2180 REMark fp_type=...
2190 REMark $Dx.xx.xx.xx.xx.xx - %binary number
2200 REMark $Ex.xx.xx.xx.xx.xx - $hex number
2210 REMark $Fx.xx.xx.xx.xx.xx - need to mask out the
first $F !
2220 :
2230 MOVE_POSITION #3, -1: REMark back up to the
first byte
2240 number$ = FETCH_BYTES(#3, 6)
2250 number$(1) = CHR$( CODE(number$(1)) && 15)
2260 POKE_STRING float_buffer, number$
2270 SElect ON fpt
2280     =1 : PRINT
'$';LTrim$(BIN$(PEEK_FLOAT(float_buffer), 32));
2290     =2 : PRINT
'$';LTrim$(HEX$(PEEK_FLOAT(float_buffer), 32));
2300     =3 : PRINT PEEK_FLOAT(float_buffer);
2310 END SElect
```


QUANTA Magazine

```
2320 END DEFine floating_points
2330 :
2340 DEFine FuNction LTrim$(str$)
2350   REMark added 22.01.10 for % and $ values - DJ
2360   REMark remove leading zeros from binary or hex
strings
2370   LOCal a,t$
2380   t$ = str$ : REMark full length by default
2390   FOR a = 1 TO LEN(t$)
2400     IF t$(a) <> '0' THEN t$ = t$(a TO LEN(t$)) :
EXIT a
2410   NEXT a
2420   t$ = '0' : REMark in case it was all zeros
2430   END FOR a
2440   RETurn t$
2450 END DEFine LTrim$
2460 :
2470 DEFine PROCedure initialise
2480   LOCal x
2490   :
2500   _quit = 0
2510   last_line_size = 0
2520   line_size = 0
2530   name_table_entries = 0
2540   :
2550   RESTORE 2580
2560   DIM keyword$(31, 9)
2570   FOR x = 1 TO 31: READ keyword$(x): END FOR x
2580   DATA 'END', 'FOR', 'IF', 'REPeat', 'SELEct',
'WHEN', 'DEFine'
2590   DATA 'PROCedure', 'FuNction', 'GO', 'TO', 'SUB',
'', 'ERRor', ''
2600   DATA '', 'RESTORE', 'NEXT', 'EXIT', 'ELSE',
'ON', 'RETurn'
2610   DATA 'REMAINDER', 'DATA', 'DIM', 'LOCal', 'LET',
'THEN', 'STEP'
2620   DATA 'REMark', 'MISTake'
2630   :
2640   DIM symbol$(10)
2650   symbol$ = '=:#,(){} ' & CHR$(10)
2660   :
```

QUANTA Magazine

```
2670 DIM operator$(22, 5)
2680 FOR x = 1 TO 22: READ operator$(x): END FOR x
2690 DATA '+', '-', '*', '/', '>=', '>', '==', '=',
'<>', '<=', '<'
2700 DATA '||', '&&', '^', '^', '&', 'OR', 'AND',
'XOR', 'MOD'
2710 DATA 'DIV', 'INSTR'
2720 :
2730 DIM monadic$(4, 3)
2740 FOR x = 1 TO 4: READ monadic$(x): END FOR x
2750 DATA '+', '-', '~', 'NOT'
2760 :
2770 DIM separator$(5, 2)
2780 FOR x = 1 TO 5: READ separator$(x): END FOR x
2790 DATA ', ', ';', '\', '!', 'TO'
2800 :
2810 END DEFine initialise
```

End of listing.

POINTS OF VIEW

Stephen Usher

Around the end of June, Malcolm Cadman's comments on the subject of "The future of the QL", prompted by an article from QUANTA Chairman, received the following reply from Stephen Usher (and others on the same topic: - Ed.

The QL Future

Malcolm Cadman wrote:

"Hi Steve,

Nice link ... :-)

Chris Curry and Clive Sinclair originally worked together, and then split with the Acorn/Sinclair rivalry (friendly though).

Coming back together with a common hardware platform - the ARM chips - would be interesting.

QUANTA Magazine

RISCOS Open is a nice idea, too, to take forward the development.

Interesting to hear that is now happening.

I still use my Archimedes ... :-)

Would that work for a QDOS/SMSQ/E "Open" ... ?

Anyway, something needs to get done. ”

[This was followed by Stephen Usher's comments very soon afterwards -Ed]

Good evening,

Indeed.... I've actually been thinking more about this over the day...

Well, I see that there are two issues with the QL legacy, one which gives a QL-like experience (i.e. SuperBASIC etc.) and emulation for running old programs.

Now, for the former, I can see a niche market just waiting to be filled.

Firstly, read this link:

http://www.osnews.com/story/23464/Why_Johnny_Can_t_Code

Then, think back to the BBC programme, Electric Dreams, the 1980s episode (unfortunately not now available to view):

<http://www.bbc.co.uk/programmes/b00n59t4>

The boy was amazed by the BBC micro, as was his friend. They loved to be able to make the machine do what THEY wanted it to do quickly and easily. (As opposed to the current crop of OSs which ALLOW the user to do what the application developers thought that the user SHOULD do and no more.)

So, I envisage the following in this case, a re-implementation of SuperBASIC, extendable but basically the same as the original, developed using a cross-platform graphics toolkit, such as QT (which runs on UNIX/Linux, MacOS and Windows and has a mobile version too, useful

QUANTA Magazine

later, see below). In its basic form, you could even have it use the raw, whole display.

This could actually lead on to a second stage, the almost instant on Linux/QL hybrid. Replace the Linux init process with this SuperBASIC interpreter (plus display driver) and the system would boot within a couple of seconds, be it ARM based or Intel it doesn't matter. Now for the clever bit... when you called EXEC or EXEC_W the program being referenced would be looked at and its type determined. If it is a QDOS program then a virtual machine would be started with a QDOS compatible OS inside and the program would be run in that. If it were a native Linux binary then it would be able to be run as well, as would a "SuperBASIC" program.

Writing the "SuperBASIC" application as a stand-alone application, running within other windowing systems should be the priority but with a thought to developing the "kiosk-mode" version for a later QL-like, (pseudo-)instant-on system.

i.e. the best of all worlds and standing on the shoulders of the Linux developers, who have done all the hard hardware work and using commodity hardware.

Thoughts?

Steve

[Now there are a few thoughts to play with. Does anyone know any men (or women) who can make things like this happen or are we destined to just sit and think about them?

I'm afraid that I either didn't see Malcolm's reply to this thread - or perhaps he didn't make one.

If you are one of those people who like to sit and think what the future holds for the humble QL, just send your musings to your favourite magazine so that we can all input some brain power to the ideas. Who knows what could come out of it? - Ed]

Send your comments to:

POINTS OF VIEW, QUANTA Magazine c/o

Editor@quanta.org.uk

You may have noticed that in some television programmes and in some films a figure appears to be transparent, like a ghost. Here are some hints as to how you can achieve this effect on a QL with GD2 colours.

Along with GD2 colours came new sprite definitions. If you look closely at these you will see that sprites can be defined to be transparent so that, as they move about the screen, you can still see the background they are covering. Not that I have seen many, or any, of these myself. But, if you do want to experiment with these you can use the S*BASIC program DO_SPRT_BAS to produce such a sprite.

However, what I am interested in now is not the sprites themselves but the mechanism by which they can be made transparent. If we knew that mechanism we would, presumably, be able to produce ghostly images in a window on the screen.

Suppose we have a background onto which we want to set a semi-transparent foreground. What can we say about this? First, both background and foreground consist of a set of pixels each of which has its own colour. The resulting composite picture will also consist of a number of pixels each with its own colour. Considering just one pixel position we see that the resulting colour must somehow depend only on the colours of corresponding pixels in background and foreground. If the foreground is to be totally transparent, the resulting pixel will have the colour of the background. If, on the other hand, the foreground is to be opaque, the resulting pixel must have the colour of the foreground. Hence, for transparencies in between, the resulting colour must somehow be a combination of the foreground colour and the background colour.

How should this be done? The colour of a pixel consists of particular intensities of red, green and blue, each amount ranging from zero to full intensity. If all colours have zero intensity the resulting colour is black. If all are at maximum the resulting colour is white. All the other 32766 colours have different proportions of the basic red, green and blue.

The obvious way of producing the resultant colour for any one pixel is thus to produce an interpolated intensity for each of the three basic colours.

QUANTA Magazine

Thus, for transparency of 50% the intensities will be the average of the intensities of foreground and background for each of red, green and blue.

Details

A pixel colour is held in a 16 bit word. The arrangement of red, green and blue intensities for mode 32, which is the mode used by QPC2, is shown here.

Colour word

Bit Number - 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
Colour Bit - g4 g5 g6 b1 b2 b3 b4 b5 r1 r2 r3 r4 r5 g1 g2 g3

The intensities of red and blue are held in five bits each: green's intensity has six bits. The bits for each colour are numbered from 1 to 5 or 6, the bits with lower numbers being the more significant.

Functions for extracting the colour values from a pixel are given now. The parameter for these functions is j%, the colour of the pixel.

```
2000 DEFine FuNction G_Red%(j%)
2010 RETurn ($F8&&j%) DIV 8
2020 END DEFine
2030 :
2040 DEFine FuNction G_Green%(j%)
2050 RETurn (7&&j%)*8+((j% DIV 8)192)&&7)
2060 END DEFine
2070 :
2080 DEFine FuNction G_Blue%(j%)
2090 RETurn ($1F00&&j%) DIV 256
2100 END DEFine
```

The calculation giving the combined value of the intensity, c', is:

$$c' = c_0 + T*(c_1 - c_0)/255$$

QUANTA Magazine

Here c_1 and c_0 are the intensities of background and foreground and T , which ranges from 0 to 255, is the transparency required.

The new values of the intensities must be combined into the new colour word. The following procedures and function do this. Here, the parameter $j\%$ is the name of the new colour word and $k\%$ is the intensity to be entered therein.

The function $Adj\%$ is needed to prevent an overflow occurring if the green bit g_4 is equal to 1. In that case the last three bits of green would have a value 4, 5, 6 or 7 and the multiplication by 2^{13} in P_Green would cause overflow. Replacing the numbers 4 to 7 by -4 to -1 prevents that.

```
2120 DEFine PROCedure P_Red(j%,k%)
2130   j%=(k%*8)^^($FFFFFF07&&j%)
2140 END DEFine
2150 :
2160 DEFine PROCedure P_Blue(j%,k%)
2170   j%=(k%*256)^^($FFFFFFE0FF&&j%)
2180 END DEFine
2190 :
2200 DEFine PROCedure P_Green(j%,k%)
2210   j%=(Adj%(k%&&7)*2^13+(k%&&$38)/8)^^($1FF8&&j%)
2220 END DEFine
2230 :
2240 DEFine FuNction Adj%(d%)
2250   IF d%>3:RETurn d%-8:ELSE RETurn d%
2260 END DEFine
```

Application

I have indicated how you can determine the value of the colour of each pixel in a combination of background and semi-transparent foreground. To produce the final result you simply have to apply this to each of the pixels. The question is, how best to do this?

Two extremely useful Traps provided by the Pointer Environment are `IOP.SVPW` and `IOP.RSPW`. These, respectively, save and restore part

QUANTA Magazine

of a window. The keywords BSVPW (function) and BRSPW (procedure) allow these alterations to be done in S*BASIC.

The syntax of the latter is:

BRSPW[#ch], buffer, bxo, byo, wszx, wszy, wxo, wyo, [, keep]

The channel, ch, is optional and defaults to #1.

"buffer" is the address of the the area holding the information, which is in PSA format.

bxo/byo is the initial position in the buffer from which the window is restored.

wszx/wszy is the size of the area in pixels.

wxo/wyo is the origin in the window of the area restored.

If "keep" is zero or not given, the area is returned to the heap.

You can use the function BSVPW to set up the saved area if you like. I prefer to use GSV_TASK which saves an area of the screen in PSA format as a file in ram1. This file can be loaded into ram by LBYTES and can be set to any window by BRSPW.

If you have loaded both foreground and background into ram by this means then you can access their pixels and produce the combined result which can, for example, simply overwrite one of background or foreground.

I show overleaf an example of a combined picture.

Appendix A

1. DO_SPRT_BAS is found in TPTRP19.ZIP on my website. An explanation on how to use it is given in SPB_TEXT.

2. The keywords BSVPW and BRSPW are part of TPTR_EXT which should be LRESPrd.

TPTR_EXT is found in TPTRP19.ZIP on my website.



3. The program GSV_TASK is in SVSCRPO6.ZIP on my website. An explanation is given in SVSCR_TEXT. After GSV_TASK is EXecuted, pressing SHIFT/TAB brings up a movable rectangle which can cover any area of the screen. Pressing ENTER will save the area chosen.

Appendix B

The format of a Partial Save Area (PSA) is:

Item	Size	Description
Spare	long	may be used by the application
Flag	word	\$4AFC
x-size	word	width in pixels
y-size	word	height in pixels
Increment	word	distance in bytes from one row to the next
Mode	byte	mode of saved image (eg mode 32)
Spare	byte	zero
Image	increment*y-size	

My website is at:

<http://web.ukonlie.co.uk/george.gwilt/>

QUO VADIS
DESIGN

Independent Information
Technology Services

www.ql-qvd.com

The screenshot shows the Quo Vadis Design website. At the top left is the logo 'QUO VADIS DESIGN' and the tagline 'Independent Information Technology Services'. To the right is the text 'QL/QDOS/SMSQ/E Software'. Below this is a navigation bar with links for 'Home', 'Products', 'Support', 'Company', and 'Contact'. A 'Welcome' section contains a message about the company's 25th anniversary and a photo of a computer workstation. A 'News' section lists two items: 'QVD QL News Blog - keep up to date' and 'Quo Vadis Design Website Launched'. A 'FEATURED PRODUCT' section displays 'QPG II' software with a 'BUY NOW!' button. The footer contains copyright information and a list of supported operating systems: 'Linux | Ubuntu | Fedora | CentOS | Debian | Debian'.

QL/QDOS/SMSQ/E Software

Bruce@ql-qvd.com

Quo Vadis Design
38 Derham Gardens
Upminster
RM14 3HA
UK

Tel: +44 (0)20 71930539
Fax: +44 (0)870 0568755

Check the QL News Blog on
our website for updates.

www.ql-qvd.com/blog



Subscriptions taken online

This program has not been fully beta-tested as Bruno Coativy is at present fully occupied with other things. It needs TK2 and resetting before running.

Some years ago now, I sent a complicated pattern-matching program to the Editor who published it in the magazine. Some time after he tried using it to detect minor changes between different versions of some articles, but unfortunately the program had a bug (but only with very large files) and kept stopping, leaving an error message. At the time I could not discover the exact nature of the problem, except that the routine was trying to construct strings in excess of 32766 characters long. But as I had made no programming notes, my own code totally baffled me and tied my brain in knots trying to unravel it. As at the time I had other more urgent projects I apologised to the Editor and admitted defeat.

Of course the main problem was that I had written the code progressively, using one-letter variables to keep the program as short as possible, and had all the code in my head. But when the Editor later wrote of his problem I had forgotten most of it. So recently I recovered some old development versions off my newly repaired Microdrives, and recovered one or two which had some REMarks, enough, that is, to be able to work out my original logic. But still the routines were too tightly packed to facilitate tweaking.

The breakthrough came when I found a version which worked uniquely on pseudo-binary files. My published program operated on any alphanumerical files, and was therefore much more complicated than need be. So I decided that my revised pattern-matching program would only work with files translated into binary, to help the random generation of strings likely to contain common elements. The simplest solution was therefore to start again from scratch, including plenty of REMarks, and in just two morning sessions I had it licked... Pseudo-Binary has the obvious benefit that it is universal, that is, it doesn't care whether it works on text, images, sound or whatever!

My new more highly optimised code gives the possibility of operating on massive files. What the routines do is to scan each file for equal sections of strings, the longest first then the remainder getting progressively smaller. It then prints out the common sections. The next

QUANTA Magazine

stage would be to attribute a statistical percentage of similarity between the two files, but I will leave that up to you so as to keep the code short. But remember to add unobtrusive padding characters to make both strings the same length...

I will not describe the routine further here, as the REMarked code should be clear enough as it stands. Just suffice it to say that it makes heavy use of tagging any recognised substrings so that subsequent use of INSTRings do not need to search unnecessarily. Indeed the more long found substrings there are, the faster the searching gets. This applies not only to the string being searched, but also to the current substring being compared. So this program is far more efficient and faster than the original.

Pattern Matching is an important part of computer theory, used by such processes as 'Search Engines' on the Internet, image recognition, identifying melodies or hunting out viruses etc. My code starts by searching for the longest possible matches, and ends with one-character ones. This program of course compares randomly-generated binary strings to demonstrate and test it, but feel free to use it with any sort of characters.

The program could be greatly improved by adapting it with subtle statistical analysis to characterise the degree of similarity, but that will have to wait for a subsequent project to be finalised. The more the compared files are similar, the faster will run the program. So the more similar your files are the longer they may be. Very large randomly-generated files will be considerably slower, so don't lengthen the demo defaults... With strings of 128 characters long, there are factorial $N!$ possible matches to test, a gigantic number which would take an inordinate length of time to complete. By masking off 'found' matched strings, the more matches you find, the less work INSTRing has to do. Masking is a very efficient element in programming, which you will do well to study and master. It neatly resolves many program search tasks.

This sort of code is generally patented before publication, but as always I shall leave it in the public domain, with the restriction that no commercial use be made of it without prior permission. (I am essentially in favour of the need to share knowledge in all spheres of Science).

So after a break of some seventeen years or so, I am back to working with statistical data, which I learnt in the mid-sixties when apprenticed as a Quality Controller... I would like to dedicate this program

QUANTA Magazine

to the memory of the late Editor, Colin Baskett, who was always very kind and helpful and without whose feedback this program would never have been fully debugged.

Best Wishes,

```
100 ::
110 REMark Match_bas by S.Poole v10may93, v21aug09,
beta-test by B.Coativy
120 CLEAR: WINDOW 512,256,0,0: CLS: A$='01101011':
b$='11011010': L=128
130 REMark Test using two pseudo-binary strings:
140 A$='': b$='': FOR f=1 TO L: A$=A$&RND(1):
b$=b$&RND(1)
150 REMark create two clone strings for masking:
160 PRINT\A$\b$: m$='': leng=LEN(A$): c$=A$: d$=b$
170 :
180 REMark Search the longest string first:
190 FOR str_len=leng TO 1 STEP -1
200     REMark start searching from the left:
210     mid_st=1
220     REMark Determine the position of the substring:
230     FOR lft_pos=mid_st TO leng-str_len+1
240         rgt_pos=lft_pos+str_len-1
250         mid$=c$(lft_pos TO rgt_pos)
260         :
270         REMark don't consider substrings
containing mask characters:
280         IF '?' INSTR mid$: GO TO 380
290         ins=mid$ INSTR d$
300         :
310         REMark If match found, add it to the
result list:
320         IF ins THEN
330             m$=m$&' '&mid$: BEEP 1234,5
340             REMark now mask off the substring in
both main strings:
350             FOR cword=lft_pos TO rgt_pos:
c$(cword)='?'
```

QUANTA Magazine

```
360             FOR dword=ins TO (ins+LEN(mid$)-1):
d$(dword)='*'
370             END IF
380         END FOR lft_pos
390         REMark now look further to the right:
400         mid_st=mid_st+1
410     END FOR str_len
420 REMark Show the matches:
430 PRINT\m$: i$=INKEY$(#1,300): RUN
440 ::
```

End of Listing

TEN YEARS AGO

John Gilpin

While browsing through some back issues looking for inspiration for some additional content for this issue of QUANTA Magazine, I came across the following snippets from QUANTA Vol. 17 Iss. 7 – (which just happens to be exactly 10 years ago – August 2000.)

SPACE FILLING

by Bill Richardson:

At a recent computer expo (comdex), Bill Gates reportedly compared the Computer Industry with the Auto Industry and stated: “If GM (General Motors) had kept up with technology like the computer industry, we would be driving twenty-five dollar cars that got 1000 miles to the gallon.”

In response to Bill’s comments, General Motors issued a press statement stating: “If GM had kept up with technology like Microsoft, we would all be driving cars with the following characteristics.

- For no reason whatsoever, your car would crash twice a day.
- Every time they repainted the lines on the road, you would have to buy a new car.
- Occasionally, your car would die on the freeway for no reason, and you would just accept this, restart and drive on.
- Occasionally, executing a manoeuvre such as a left turn would cause your car to shut down, in which case you would have to reinstall the engine.

QUANTA Magazine

- Only one person at a time could use the car, unless you bought “car95” or “carnt”. But then you would have to buy more seats.
- Macintosh would make a car that was powered by the sun, reliable, five times as fast, and twice as easy to drive, but would only run on five per cent of the roads.
- The oil, water temperature and alternating warning lights would be replaced by a single “general car fault” warning light.
- New seats would force everyone to have the same size bottoms.
- The airbag system would say “are you sure?” before going off.
- Occasionally, for no reason whatsoever, your car would lock you out, and refuse to let you in, until you simultaneously lift the door handle, turn the key and grab hold of the radio antenna.
- GM would require all car buyers to also purchase a deluxe set of Rand-McNally road maps (now a GM subsidiary), even though they neither need them nor want them. Attempting to delete this option would immediately cause the car’s performance to diminish by 50% or more. Moreover, GM would become a target for investigation by the justice department.
- Every time GM introduced a new model, car buyers would have to learn to drive all over again because none of the controls would operate in the same manner as in the old car.
- You’d have to press the “Start” button to shut off the engine.

With acknowledgements to the originator of this piece, unknown to anyone at this time

Bill Richardson.

And on a more serious note, this tip to all Qlers was penned by our late Editor from August 2000, Colin Baskett:

XCHANGE QUILL TIP

by: Colin Baskett

Have you ever had to COPY something from the end of a long Quill document to the beginning? You can’t even make a cup of coffee because you have to keep your finger on the cursor key.

All that is a thing of the past for Xchange users thanks to the EXTRACT command.

It really is dead easy. Select the passage to be copied – just as with the COPY command. Choose whether to keep or to delete the existing

QUANTA Magazine

passage. Transfer to “buffer” (a file – buffer_doc – on the default drive) or to another named file. End of EXTRACT command. GOTO top of document (or wherever) and use the MERGE command. In XCHANGE Quill the default MERGE procedure is to merge in the contents of the buffer. Effortless and quick.

Of course, there are snags. There always are:

<F3, F3, E> for EXTRACT is all too readily confused with <F3, E> for ERASE with potentially disastrous consequences.

Less common – but equally infuriating – is to get side-tracked in the middle of such a transfer process with your precious text residing in the buffer and, then, absent-mindedly to set about some trivial text transfer again using EXTRACT. I know, I’ve done it – and more than once!

EXTRACT does not explicitly warn you that you are about to overwrite the contents of the buffer (a serious shortcoming) even though it does give you the alternative of APPENDING when there is already something in the buffer.

Now the APPEND alternative is really quite useful. It enables you to go through a document extracting bits and pieces to string them together, possibly as a list or even as a connected piece of text.

Again, The same snag exists that the pre-existing contents of the buffer will be lost if you REPLACE instead of APPENDING. All too easily done. Just press ENTER once too often, but then that’s life.

It is not too difficult to work out safety routines. The best safe-guard is, of course, to save your text before using EXTRACT and to backup the buffer frequently when accumulating a number of appended “extracts” in it. However, I still occasionally lose text when using EXTRACT but then so I do with COPY.

To sum up, EXTRACT is a useful and versatile facility which makes Quill significantly more user friendly. It is worth getting XCHANGE just for the sake of EXTRACT.

Colin Baskett.

And finally from the August 2000 archive:



Independent QL Users Group

World-wide Membership is by subscription only,

Offering the following benefits:

Bimonthly Magazine - up to 52 pages

Massive Software Library - All Free!

Free Helpline and Workshops

Regional Sub-Groups. One near you?

Advice on Software and Hardware problems

Subscription just £14 for Full Membership

PayPal (see QUANTA Web Site),

Cash, Cheques and Postal Orders Accepted

Now in our Twenty Seventh Year

Further details from the Membership Secretary

**John Gilpin, 181, Urmston Lane,
Stretford, Manchester, M32 9EH (UK).**

Tel. +44 (0) 161 865 2872

Email: membership@quanta.org.uk

Visit the QUANTA Web Site

<http://www.quanta.org.uk>

QUANTA extends a very warm welcome to Bruce Nicholls, the new Co-Editor of QL Today. Bruce has been a new software author, a software publisher (Quo Vadis Design) and a magazine editor (QReview). Welcome back, Bruce!

Dilwyn Jones continues as Editor but says that with his new day-job (something to do with the Snowdon Mountain Railway, I believe) he had been struggling to do justice to QL Today – which is I suppose Dilwyn's night-job. Dilwyn! I have long wondered how on earth you found time to do all you have done.

[And he still continues to churn out the work, month by month, ten years later - Thank you Dilwyn - Ed.]

SUBGROUP MEETINGS

THE WEST MIDLANDS QL USER GROUP

The new meeting place is called the Bull, the address is 1 Price Street, Birmingham, B4 6JU. approx 7 minutes walk from Snow Hill main-line railway station and has ample street parking nearby. Meetings remain open to QUANTA members and non-members alike.

After your first meeting you're expected to pay £1 each time towards the organisation and monthly newsletter costs - except once a year when the group pays all those who attend a pound instead ;-)

We provide food twice a year for those who attend the A.G.M. and the group's birthday party and at meetings we discuss all sorts of things, QL-related and otherwise, usually from about 8:20pm onwards till 10:30 or later. The group owns QL hardware, a disk library and subscribes to relevant magazines. People can join the mailing list for £3.50 per year, even if they can't attend regularly.

Meetings : Meetings are usually on the first and third Monday of each month (but not on bank holidays unless the previous one was cancelled for that reason).

Contact : The contact number remains Mike Bedford White's, on 0121 605 4438, anytime after 11am.

Email: mikebwhite@blueyonder.co.uk

QUANTA Magazine

QUANTA DORSET SUBGROUP

Meetings are held, on the second Sunday of each month at the Merley Community Centre in Harrier Drive, Merley, just south of Wimborne Minster. Time: 2.00 pm. to 5.00 pm. All are welcome.

John Mason, 01425 275894.

LONDON QL AND QUANTA GROUP

Meetings are held in the School Room, which is the basement of the Borough Welsh Congregational Chapel, 90, Southwark Bridge Road, London SE1. This is almost opposite the junction with Marshalsea Road, at the other end of which, 5 minutes walk away, is the Borough Underground Station. Free parking is easily available in Southwark Bridge Road for road users, Cyclists are welcome.

Members currently each pay a subscription of £45 per year towards the cost of the hire of the hall and a small additional charge is made for tea/coffee - biscuits are then provided free.

Please bring your QL equipment if possible, all types welcome, e.g., PC portable running emulators. The Group has some systems stored at the venue for its use. In addition, there are lots of donated second user equipment, books, magazines, etc, available for purchase.

The London Group is moving over from 10 Meetings a Year, to 8 Meetings a Year.

In 2010 the Group will take a two month break in July and August, and then resume Meetings in September.

The Group will not meet in December 2010.

Therefore the next 3 Meetings will be - September, October and November.

The details of the new 8 Meetings scheduled for 2011, will be announced nearer the time.

Malcolm Cadman: Tel: 020 8691 5780

email: QL@mcad.demon.co.uk

QUANTA Magazine

NEMQLUG - THE NORTH EAST MANCHESTER QL USER GROUP

Meetings are held at 181, Urmston Lane, Stretford, Manchester. M32 9EH (Near Junction 7 of M60 Motorway - Was Junction 7 M63 Motorway). Our informal meetings start at 7 p.m. and finish at 11 pm. All welcome.

Meets on the last Thursday of each month (except December).

John/Sarah Gilpin 0161-865 2872 (thegilpins@btopenworld.com)

SCOTTISH QL USERS GROUP

SQLUG meets at George Gwilt's home every second Sunday in the month. Please contact George Gwilt before coming to get the location.

Gdgqler@gmail.com.

SOLENT SUBGROUP

Meets at Botley Market Hall, all welcome. Park at the back and use side entrance.

1st Saturday of the month from 2.00 pm to 6.00 pm.

Graham Evans, Tel: 023 8040 3350.

SURREY QUANTA SUBGROUP (SQSG)

The group is currently trying other venues in the Leatherhead - Epsom area. If you plan to attend but do not get our emails, please phone or email Ken Bain for the latest location.

Meetings are 8.00 to 10.00 pm. On the last Wednesday of each month except December.

Ken Bain - 01932 - 347 432 (to midnight), kenb@bcs.org.uk